EXAMINER'S AMENDMENT

1. An examiner's amendment to the record appears below. Should the changes and/or additions be unacceptable to applicant, an amendment may be filed as provided by 37 CFR 1.312. To ensure consideration of such an amendment, it MUST be submitted no later than the payment of the issue fee.

2. Authorization for this examiner's amendment was given in a telephone interview with Mr. Xiaomin Huang on01/26/2010.

3. Amend the claims as the following paragraph:

In the Claims

Pending claims:

1-11. (Canceled).

12. (Previously Presented) A method comprising:

providing a connection class to include generic routines to connect to peripheral devices, the connection class to be independent of device-specific features of the peripheral devices;

providing a plurality of parameters to define specific features of the peripheral devices;

receiving a request to access one of the peripheral devices;

determining whether the requested peripheral device is accessible;

if the request is a request to connect a computer to the requested peripheral device and the requested peripheral device is accessible,

instantiating the connection class to create an object specific to the requested peripheral device,

using the instantiated object to cause a native driver of the requested peripheral device to execute, and

connecting, through the native driver, the computer to the requested peripheral device based upon the parameters of the requested peripheral device; and

if the request is a request from the requested peripheral device to send data to the computer,

notifying the computer that the requested peripheral device has the data,

instantiating the connection class to create an object specific to the requested peripheral device,

using the instantiated object to cause the native driver of the requested peripheral device to execute,

connecting, through the native driver, the computer to the requested peripheral device; and

sending the data from the requested peripheral device to the computer.

13. (Previously Presented) The method of claim 12, further comprising:

if the request is a request to disconnect the computer from the requested peripheral device,

using the instantiated object of the connection class to cause the native driver of the requested peripheral device to execute,

disconnecting, through the native driver, the computer from the requested peripheral device, and

uninstantiating the connection class to delete the instantiated object.

14-21. (Canceled).

22. (New) The method of claim 12, wherein the peripheral devices are selected from a group consisting of a printer, a scanner, an imager, a smart card reader, and a barcode reader.

23. (New) The method of claim 12, further comprising:

providing an emulator to simulate access to the peripheral devices in order to test the connection class.

24. (New) The method of claim 12, further comprising:

providing a graphical user display to allow a user to select the peripheral devices to be accessible by the connection class; and

providing native drivers corresponding to the selected peripheral devices.

25. (New) The method of claim 12, wherein the connection class includes input and output interfaces that further include serial and parallel connection interfaces to communicate with the peripheral devices.

26. (New) A system comprising:

one or more peripheral devices having associated therewith native drivers;

a memory device storing a plurality of parameters to define specific features of the one or more peripheral devices and

a mobile computer configured to:

provide a connection class to an application, the connection class to include generic routines to connect to the one or more peripheral devices, the connection class to be independent of device-specific features of the one or more peripheral devices;

provide a plurality of parameters to define specific features of the one or more peripheral devices;

receive a request to access one of the one or more peripheral devices;

determine whether the requested peripheral device is accessible;

if the request is a request to connect the mobile computer to the requested peripheral device and the requested peripheral device is accessible,

instantiate the connection class to create an object specific to the requested peripheral device,

use the instantiated object to cause a native driver of the requested peripheral device to execute, and

connect, through the native driver, the computer to the requested peripheral device based upon the parameters of the requested peripheral device; and

if the request is a request from the requested peripheral device to send data to the mobile computer,

notify the mobile computer that the requested peripheral device has the data,

instantiate the connection class to create an object specific to the requested peripheral device,

use the instantiated object to cause the native driver of the requested peripheral device to execute,

connect, through the native driver, the computer to the requested peripheral device; and

send the data from the requested peripheral device to the computer.

27. (New) The system of claim 26, further comprising:

a second mobile computer, having the application ported thereto, configured to access a different peripheral device with the application, wherein the application on the second mobile computer uses the connection class to access the different peripheral device without modifying the application.

28. (New) The system of claim 26, wherein the mobile computer uses the connection class to limit communication with the one or more peripheral devices to one request at a time.

29. (New) The system of claim 26, wherein the peripheral devices are selected from a group consisting of a printer, a scanner, an imager, a smart card reader, and a barcode reader.

30. (New) The system of claim 26, wherein the mobile computer is further configured to:

provide an emulator to simulate access to the peripheral devices in order to test the connection class.

31. (New) The system of claim 26, wherein the mobile computer is further configured to:

provide a graphical user display to allow a user to select the peripheral devices to be accessible by the connection class; and

provide native drivers corresponding to the selected peripheral devices.

32. (New) The system of claim 26, wherein the connection class includes input and output interfaces that further include serial and parallel connection interfaces to communicate with the peripheral devices.

33. (New) Computer readable storage medium storing thereon program instructions that, when executed, cause an executing device to perform the steps of:

providing a connection class to include generic routines to connect to peripheral devices, the connection class to be independent of device-specific features of the peripheral devices;

providing a plurality of parameters to define specific features of the peripheral devices;

receiving a request to access one of the peripheral devices;

determining whether the requested peripheral device is accessible;

if the request is a request to connect a computer to the requested peripheral device and the requested peripheral device is accessible,

instantiating the connection class to create an object specific to the requested peripheral device,

using the instantiated object to cause a native driver of the requested peripheral device to execute, and

connecting, through the native driver, the computer to the requested peripheral device based upon the parameters of the requested peripheral device; and

if the request is a request from the requested peripheral device to send data to the computer,

notifying the computer that the requested peripheral device has the data,

instantiating the connection class to create an object specific to the requested peripheral device,

using the instantiated object to cause the native driver of the requested peripheral device to execute,

connecting, through the native driver, the computer to the requested peripheral device; and

sending the data from the requested peripheral device to the computer.

34. (New) The computer readable medium of claim 33, further comprising instructions that cause the executing device to perform the steps of:

if the request is a request to disconnect the computer from the requested peripheral device,

using the instantiated object of the connection class to cause the native driver of the requested peripheral device to execute,

disconnecting, through the native driver, the computer from the requested peripheral device, and

uninstantiating the connection class to delete the instantiated object.

35. (New) The computer readable medium of claim 33, wherein the peripheral devices are selected from a group consisting of a printer, a scanner, an imager, a smart card reader, and a barcode reader.

36. (New) The computer readable medium of claim 33, further comprising instructions that cause the executing device to perform the steps of:

providing an emulator to simulate access to the peripheral devices in order to test the connection class.

37. (New) The computer readable medium of claim 33, further comprising instructions that cause the executing device to perform the steps of:

providing a graphical user display to allow a user to select the peripheral devices to be accessible by the connection class; and

providing native drivers corresponding to the selected peripheral devices.

38. (New) The computer readable medium of claim 33, wherein the connection class includes input and output interfaces that further include serial and parallel connection interfaces to communicate with the peripheral devices.

Conclusion

Any inquiry concerning this communication or earlier communications from the examiner should be directed to PHUONG N. HOANG whose telephone number is (571)272-3763. The examiner can normally be reached on Monday - Friday 9:00 am to 5:30 pm.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Hyunh S. Sough can be reached on 571-272-6799. The fax phone number for the organization where this application or proceeding is assigned is 571-273-8300.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see http://pair-direct.uspto.gov. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free). If you would like assistance from a USPTO Customer Service Representative or access to the automated information system, call 800-786-9199 (IN USA OR CANADA) or 571-272-1000.

/Hyung S. Sough/
Supervisory Patent Examiner, Art Unit 2194
01/31/10

/P. N. H./
Examiner, Art Unit 2194